

AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer-readable medium having a base generator class stored thereon ~~for use by developers to create generators to perform specific~~ that provides incrementation capability that allows developers to create a new generator class for generating a generator that repeatedly performs a single operation that generates an object while having the base generator class vary a generator property of the generated object for each iteration of the single operation such that the developers need not provide code within the new generator class to perform the incrementation capability tasks, the base generator class comprising:

a base generator class constructor ~~for initializing a generator~~ that is overridden by a public default constructor of the new generator class that inherits from the base generator class, the public default constructor for initializing a generator;

a method that is overridden by an object generation method of the new generator class, the object generation method defining the single operation that generates an object at each iteration of the single operation;

a generator properties class that provides incrementation capability, which allows the value of ~~a~~ a the generator property of each generated object to vary during consecutive executions of ~~a generator~~ the object generation method of the new generator class, such that the new generator class need not provide incrementation capability, the value of a generator property comprising a stream portion and a numerical portion, the numerical portion being incremented;

a status indicator including a status user interface (UI) for displaying the execution status of generators, the execution status of each generator including a description of the generator and the value of each property associated with the generator;

a schedule class; and

a logging class for recording an object generated by each generator, a time of the object generation, and generator properties used to generate the object; the logging class providing a user a capability to turn the logging class off;

wherein the logging class is used to verify the tasks performed by the generators.

2. (Previously presented) The computer-readable medium of Claim 1, wherein the generator properties class that provides incrementation capability includes a plurality of generator properties.

3. (Original) The computer-readable medium of Claim 2, wherein said plurality of generator properties includes:

- a value of a generator property;
- a plurality of incrementation settings;
- a default incrementor that changes the value of the generator property; and
- a default validator that validates the value of the generator property.

4. (Canceled)

5. (Previously presented) The computer-readable medium of Claim 1, wherein the schedule class comprises:

- a start condition under which the execution of a generator may be started;
- a recurrence condition under which the execution of a generator may recur;
- an end condition under which the execution of a generator stops; and
- a dialog box that can be used to accept user input.

6. (Previously presented) The computer-readable medium of Claim 1, wherein the logging class enables the recording of the execution process of a generator.

7. (Currently Amended) A method, performed by a computer having a processor, of
executing~~creating~~ a generator using a base generator class to provide incrementation capability
that allows developers to create a new generator class for generating a generator that repeatedly
performs a single operation that generates an object while having the base generator class vary a
generator property of the generated object for each iteration of the single operation such that the
developers need not provide code within the new generator class to perform the incrementation
capability, wherein the generator performs a specific task including at least creating a file, the
method comprising:

creating a new generator class that inherits a base generator class that contains
incrementation capability;

~~creating~~executing, by the processor, a public default constructor for the new generator
class that overrides the base generator class constructor by accepting user-defined properties for
the generator, the user-defined properties for the generator including incrementation settings for
a first property;

executing, by the processor, an object generation method of the new generator class that
overrides a method of the base generator class, the object generation method defining the single
operation that generates an object at each iteration of the single operation, wherein the object
generated by a first execution of the object generation method includes a first value of the first
property; and

executing, by the processor, a default incrementor of the base generator class to
increment the value of the first property such that upon a second execution of the object
generation method, an object generated by the second execution includes the incremented value
of the first property.

~~implementing a function in the new generator class to perform the specific task; and~~
~~verifying the task based on properties of the generator.~~

8. (Currently Amended) The method of Claim 7, wherein ~~creating~~ executing a public
default constructor comprises:

initializing the base generator class constructor with the name and the description of the
generator; and

defining the properties of the generator.

9. (Original) The method of Claim 8, wherein defining properties for the generator comprises:

- (a) defining the name of a property;
- (b) setting a default value for the property;
- (c) providing a description for the property;
- (d) specifying incrementation settings for the property;
- (e) creating a custom property incrementor, if applicable;
- (f) creating a custom property validator, if applicable; and
- (g) repeating (a)-(f) for all properties of the generator.

10. (Currently Amended) The method of Claim 7, further comprising ~~implementing a function to be executed before each execution of a generator~~executing a method before each execution of the object generation method.

11. (Currently Amended) The method of Claim 7, further comprising ~~implementing a function to be executed after each execution of a generator~~executing a method after each execution of the object generation method.

12. (Currently Amended) A method, performed by a computer having a processor, of
~~using~~ executing a generator using a base generator class to provide incrementation capability that
allows developers to create a new generator class for generating a generator that repeatedly
performs a single operation that generates an object while having the base generator class vary a
generator property of the generated object for each iteration of the single operation such that the
developers need not provide code within the new generator class to perform the incrementation
capability that performs a specific task including at least creating a file, the method comprising:

executing, by the processor, a public default constructor for the new generator class that
overrides a base generator class constructor by accepting user-defined properties for the
generator, the user-defined properties for the generator including incrementation settings for a
first property;

customizing the incrementation settings of the generator, the incrementation settings
including incrementation settings that specify how the value of a generator property may vary
between generated objects, the generated object including a file, the incrementation settings
including at least one of an "offset" setting and a "step" setting, the "offset" setting specifying a
value by which the value of a generator property is incremented, the "step" setting specifying a
number of generated objects containing the value of a generator property with the same "offset"
setting;

executing, by the processor, an object generation method of the new generator class that
overrides a method of the base generator class, the object generation method defining the single
operation that generates the generated object at each iteration of the single operation, wherein the
object generated by the first execution of the object generation method includes a first value of
the generator property; and

executing, by the processor, a default incrementor of the base generator class to
increment the value of the generator property such that upon executing the object generation
method a number of times defined by the "step" setting, the object generated by the execution
includes a value of the generator property that is equal to the first value incremented by the value
of the "offset" setting.

~~executing the generator with the customized settings; and~~
~~verifying the task based on the settings of the generator.~~

13. (Currently Amended) The method of Claim 12, wherein customizing the incrementation settings of the generator, is accomplished through a user interface.

14. (Previously Presented) The method of Claim 13, further comprising:
starting an object generator user interface;
selecting the generator; and
customizing properties of the generator.

15. (Currently Amended) The method of Claim 14, wherein selecting ~~a~~ the generator further ~~comprising~~ comprises adding the generator from files containing one or more generators.

16. (Currently Amended) The method of Claim 14, further comprising loading the settings of ~~a~~ the generator from a file.

17. (Previously Presented) The method of Claim 14, wherein customizing the properties of the generator comprises:

- (a) selecting one property;
- (b) specifying a value of the one property;
- (c) specifying incrementation settings of the one property; and
- (d) repeating (a)-(c) until there are no more properties to be customized.

18. (Original) The method of Claim 14, further comprising setting a schedule for executing the generator.

19. (Original) The method of Claim 14, further comprising setting logging options for executing the generator.

20. (Original) The method of Claim 14, further comprising saving the settings of the generator.

21. (Currently Amended) The method of Claim 12, wherein customizing the

incrementation settings of a generator is accomplished programmatically.

22. (Previously presented) The method of Claim 21, further comprising:
creating a new instance of the generator;
setting a number of objects to be generated by the generator; and
customizing properties of the generator.
23. (Previously presented) The method of Claim 22, wherein customizing properties of the generator comprises:
- (a) setting values of the properties; and
 - (b) specifying incrementation settings of the properties.
24. (Original) The method of Claim 21, further comprising:
creating a new instance of the generator; and
loading saved settings of the generator from a file.
25. (Original) The method of Claim 21, further comprising:
creating a new instance of the generator;
loading saved settings of the generator from a file; and
implementing a function to execute the generator asynchronously.
26. (Original) The method of Claim 21, further comprising:
creating a new instance of the generator;
loading saved settings of the generator from a file;
displaying an object generation status UI; and
adding the generator to the object generation status UI.
27. (Original) The method of Claim 21, further comprising:
creating a new instance of the generator;
loading saved settings of the generator from a file;
displaying a schedule dialog box that allows a user to specify a schedule for executing the

generator; and

displaying a logging dialog box that allows a user to specify logging options for executing the generator.

28. (Original) The method of Claim 12, further comprising executing the generator through a user interface.

29. (Original) The method of Claim 12, further comprising executing the generator programmatically.

30-31. (Canceled)

32. (Currently Amended) The computer-readable medium of claim 57~~method of Claim 31, creating a public default constructor further comprising~~wherein the public default constructor is generated by:

initializing the base generator class constructor with the name and the description of the generator; and

defining the properties of the generator.

33. (Currently Amended) The computer-readable medium~~method~~ of Claim 32, wherein defining the properties for the generator includes:

- (a) defining names of the properties;
- (b) setting default values for the properties;
- (c) providing descriptions for the properties;
- (d) specifying incrementation settings for the properties;
- (e) creating a custom property incrementor, if applicable; and
- (f) creating a custom property validator, if applicable.

34. (Currently Amended) The computer-readable medium of claim 57~~method of Claim 31~~, further comprising

implementing-executing a function ~~to be executed~~ before each execution of the

generator.

35. (Currently Amended) The computer-readable medium of claim 57~~method of Claim 31~~, further comprising
implementing executing a function to be executed after each execution of the generator.

36. (Currently Amended) The computer-readable medium of claim 57~~method of Claim 30~~, wherein customizing the incrementation settings of a generator is accomplished through a user interface.

37. (Currently Amended) The computer-readable medium~~method~~ of Claim 36, further comprising:
starting an object generator user interface;
selecting the generator; and
customizing properties of the generator.

38. (Currently Amended) The computer-readable medium~~method~~ of Claim 37, wherein selecting the generator comprises adding the generator from files containing one or more generators.

39. (Currently Amended) The computer-readable medium~~method~~ of Claim 37, further comprising loading generator settings from a file.

40. (Currently Amended) The computer-readable medium~~method~~ of Claim 37, wherein customizing the properties of the generator comprises:

- (a) selecting one property;
- (b) specifying the value of the one property;
- (c) specifying incrementation settings of the one property; and
- (d) repeating (a)-(c) until there are no more properties to be customized.

41. (Currently Amended) The computer-readable medium ~~method~~ of Claim 37,

further comprising setting a schedule for executing the generator.

42. (Currently Amended) The computer-readable medium~~method~~ of Claim 37, further comprising setting logging options for executing the generator.

43. (Currently Amended) The computer-readable medium~~method~~ of Claim 37, further comprising saving the settings of the generator.

44. (Currently Amended) The computer-readable medium of claim 57~~method of Claim 30~~, wherein customizing the incrementation settings of a generator is accomplished programmatically.

45. (Currently Amended) The computer-readable medium ~~method~~ of Claim 44, further comprising:

- creating a new instance of the generator;
- setting a number of objects to be generated by the generator; and
- customizing the properties of the generator.

46. (Currently Amended) The computer-readable medium ~~method~~ of Claim 45, wherein customizing the properties of the generator comprises:

- (a) setting a value of one property;
- (b) specifying incrementation settings of the one property; and
- (c) repeating (a)-(b) until there are no more properties to be customized.

47. (Currently Amended) The computer-readable medium ~~method~~ of Claim 44, further comprising:

- creating a new instance of the generator; and
- loading saved settings of the generator from a file.

48. (Currently Amended) The computer-readable medium ~~method~~ of Claim 44,

further comprising:

- creating a new instance of the generator;
- loading saved settings of the generator from a file; and
- implementing a method to execute the generator asynchronously.

49. (Currently Amended) The computer-readable medium ~~method~~ of Claim 44, further comprising:

- creating a new instance of the generator;
- loading saved settings of the generator from a file;
- displaying an object generation status UI; and
- adding the current generator to the object generation status UI.

50. (Currently Amended) The computer-readable medium ~~method~~ of Claim 44, further comprising:

- creating a new instance of the generator;
- loading saved settings of the generator from a file;
- displaying a schedule dialog box that allows a user to specify a schedule for executing the generator; and
- displaying a logging dialog box that allows a user to specify logging options for executing the generator.

51. (Currently Amended) The computer-readable medium of claim 57 ~~method~~ of Claim 30, further comprising executing the generator through a user interface.

52. (Currently Amended) The computer-readable medium of claim 57 ~~method~~ of Claim 30, further comprising executing the generator programmatically.

53-56. (Canceled)

57. (New) A computer-readable medium having computer executable instructions which when executed by a processor of a computer perform a method of executing a generator using a base generator class to provide incrementation capability that allows developers to create a new generator class for generating a generator that repeatedly performs a single operation that generates an object while having the base generator class vary a generator property of the generated object for each iteration of the single operation such that the developers need not provide code within the new generator class to perform the incrementation capability, the method comprising:

executing, by the processor, a public default constructor for the new generator class that overrides a base generator class constructor by accepting user-defined properties for the generator, the user-defined properties for the generator including incrementation settings for a first property;

customizing the incrementation settings of the generator, the incrementation settings including incrementation settings that specify how the value of a generator property may vary between generated objects, the generated object including a file, the incrementation settings including at least one of an "offset" setting and a "step" setting, the "offset" setting specifying a value by which the value of a generator property is incremented, the "step" setting specifying a number of generated objects containing the value of a generator property with the same "offset" setting;

executing, by the processor, an object generation method of the new generator class that overrides a method of the base generator class, the object generation method defining the single operation that generates the generated object at each iteration of the single operation, wherein the object generated by the first execution of the object generation method includes a first value of the generator property; and

executing, by the processor, a default incrementor of the base generator class to increment the value of the generator property such that upon executing the object generation method a number of times defined by the "step" setting, the object generated by the execution includes a value of the generator property that is equal to the first value incremented by the value of the "offset" setting.